

Application Serial Number: 10/632,767

Attorney Docket Number: MS1-1541US

To: Examiner Biagini  
Fax: 571-273-9743  
Phone: 571-272-9743

From: Beatrice L. Koempel-Thomas (Reg. No. 58213)  
Lee & Hayes, PLLC  
bea@leehayes.com  
(Tel. 509-324-9256; Fax 509-323-8979)

### Agenda and Request for an Examiner Interview

– INFORMAL COMMUNICATION – FOR DISCUSSION PURPOSES ONLY –

**[0001]** This communication provides an agenda for an interview of this matter. My assistant will be contacting you to schedule an interview. If you would prefer to schedule the interview, then please contact my assistant or me directly. Our contact information is on the signature page of this document. Thank you in advance for agreeing to discuss this matter.

#### **Interview Agenda:**

- Discussion of exemplary differences between the application/claims and the cited documents
- Discussion of proposed amendments

Serial No.: 10/632,767  
Atty Docket No.: MS1-1541US  
Atty/Agent: Beatrice L. Koempel-Thomas

-1-

lee&hayes The Business of IP®

www.leehayes.com • 509.324.9256

## Exemplary Differences

**[0002]** **Claim 1** recites (with proposed amendment and emphasis added):

A method comprising:

receiving, at a client device, a plurality of **temporally non-contiguous portions of a streaming media file**, wherein:

**temporally non-contiguous portions comprise portions of a received streaming media file that are not adjacent to one another in terms of the temporal presentation of their content during playback,** and

at least a first and a second of the temporally non-contiguous portions of the received streaming media file being encoded at different bit rates; and

storing the plurality of temporally non-contiguous portions of the received streaming media file in a single cache file on the client device.

**[0003]** In contrast, it appears to me that Chou describes "two different data streams," (c. 3, ll. 29-30), which appear to be two separate files. It does not appear that Chou describes at least two "temporally non-contiguous portions of the received streaming media file ... compris[ing] portions of a received streaming media file that are not adjacent to one another in terms of the temporal presentation of their content during playback" as recited in the claim.

**[0004]** Additionally, it appears to me that Lee lacks the emphasized feature recited in claim 1, with the proposed amendment. It appears that Lee describes selective caching designed to reduce bandwidth requirements between a central server and a proxy server (c. 3-4, ll. 60-57). While Fig. 3A illustrates "stretching the delivery schedule ... over unused time slots," (c. 11, ll. 50-65). It also appears as though the caching described in Lee occurs at the proxy server (c. 3-4, ll. 67-6).

**[0005]** **Claim 3**, depending from claim 1 recites (with emphasis added):

A method as defined in claim 1, wherein the **first and second non-contiguous portions comprise video data** and wherein **a third non-contiguous portion comprises audio data**.

**[0006]** In contrast, it appears to me that Chou describes “[a]udio [and] video ... data types ... stored on [a] video server ... and delivered by an application on-demand over [a] network ... to one or more video clients,” (c. 6, ll. 60-64). And, Chou goes on to describe, “[i]n operation, ... a user clicks on a link to a video clip ... , and an application program for viewing streamed multimedia files launches from a hard disk of the video client .... . The application begins loading in a file for the video which is being transmitted across the network ... from the video server,” (c. 7, ll. 9-15).

**[0007]** It does not appear that Chou describes a “first and second non-contiguous portions comprise video data and wherein a third non-contiguous portion comprises audio data” as recited in the claim.

**[0008]** Additionally, it appears that Lee lacks the emphasized feature recited in claim 3. It appears that Lee describes a “media delivery system” generally applicable to “video streaming signals [and] audio streaming signals,” (c. 5, ll. 44-49). The Office Action also cites column 7, lines 55-58. However, it is not clear how Lee is being interpreted to allegedly anticipate claim 3.

**[0009] Regarding claim 5,** the Office Action relies on Lee in combination with Pinckney in leveling the rejection, however Pinckney describes a "Streaming Delivery Accelerator (SDA) ... intermediate between the content provider ... and one or more clients, (Pinckney [0031]). And, Pinckney explicitly describes, "the SDA receives a source content file 910 from a content provider and 'shreds' the source content file 910 into ... contiguous files 920, 930, 940 [that] represent contiguous subsets of the content file 910," (Pinckney [0055]-[0058]). Thus, it appears that 920, 930, and 940 of Pinckney are contiguous and that Pinckney lacks at least the non-contiguous portions as recited in the claim.

**[00010]** I would also like to discuss the basis of rejection of **claims 15 and 18**, depending from claim 11, based on Lee in view of Pinckney [0054]-[0058].

**[00011]** If time permits, I would like to discuss the basis of rejection of **claim 23**, formerly depending from claim 21 (proposed incorporated into claim 21), based on Lee in view of Pinckney [0053]-[0058] and Fig. 10.

### **Proposed Amendments**

**[00012]** Please see the attached Appendix of Proposed Claim Amendments. I would like to discuss your opinion regarding the proposed amendments in light of the currently cited documents. Applicant is receptive to considering other amendments as appropriate in light of our discussion of examples of distinctions, above.

Serial No.: 10/632,767  
Atty Docket No.: MS1 -1541US  
Atty/Agent: Beatrice L. Koempel-Thomas

-4-

lee&hayes The Business of IP®

[www.leehayes.com](http://www.leehayes.com) • 509.324.9256

[00013] Thank you in advance for scheduling time for this interview. I look forward to discussing this matter with you.

Respectfully Submitted,

Lee & Hayes, PLLC  
Representative for Applicant

Beatrice L. Koempel-Thomas  
(bea@leehayes.com; 509-944-4759)  
Registration No. 58213

Assistant: Cherri A. Simon  
509-944-4776  
cherri@leehayes.com

Serial No.: 10/632,767  
Atty Docket No.: MS1 -1541US  
Atty/Agent: Beatrice L. Koempel-Thomas

-5-

lee&hayes The Business of IP®

[www.leehayes.com](http://www.leehayes.com) • 509.324.9256

**Appendix of Claims with Proposed Amendments**  
**(Withdrawn Claims Omitted)**

1.      **(Proposed Amended)**      A method comprising:  
receiving, at a client device, a plurality of temporally non-contiguous portions of a streaming media file, wherein:  
  
temporally non-contiguous portions comprise portions of a received streaming media file that are not adjacent to one another in terms of the temporal presentation of their content during playback, and  
  
at least a first and a second of the temporally non-contiguous portions of the received streaming media file being encoded at different bit rates; and  
  
storing the plurality of temporally non-contiguous portions of the received streaming media file in a single cache file on the client device.
2.      **(Original)**      A method as defined in claim 1, wherein the first and second non-contiguous portions comprise video data.
3.      **(Original)**      A method as defined in claim 1, wherein the first and second non-contiguous portions comprise video data and wherein a third non-contiguous portion comprises audio data.
4.      **(Original)**      A method as defined in claim 1, wherein the cache file is stored in non-volatile memory.

5. **(Original)** A method as defined in claim 1, wherein the act of storing comprises:

creating a plurality of media cache streams, each media cache stream being associated with a unique bit rate;

storing the first non-contiguous portion in a media cache stream associated with the bit rate of the first non-contiguous portion;

storing the second non-contiguous portion in a media cache stream associated with the bit rate of the second non-contiguous portion; and

storing the media cache streams in the cache file.

6. **(Original)** A method as defined in claim 1, wherein the act of storing comprises:

creating a first media cache stream associated with the bit rate of the first non-contiguous portion;

storing the first non-contiguous portion in a media cache segment of the first media segment stream;

creating a second media cache stream associated with the bit rate of the second non-contiguous portion;

storing the second non-contiguous portion in a media cache segment of the second media cache stream;

creating a byte cache index segment and a byte cache data segment for each media cache segment; and

storing the byte cache index segments and the byte cache data segments in the cache file.

**7. (Proposed Amended)** A method comprising:

creating a plurality of media cache streams, each media cache stream being associated with a unique bit rate;

receiving a plurality of temporally non-contiguous portions of a streaming media file, wherein:

each temporally non-contiguous portion being associated with a unique temporal section of the streaming media file;

storing each temporally non-contiguous portion in a media cache segment of a media cache stream associated with a bit rate at which the temporally non-contiguous portion was encoded, at least two of the temporally non-contiguous portions being stored in media cache segments in different media cache streams;

storing each of the media cache streams in a single cache file.

**8. (Original)** A method as defined in claim 7, wherein the act of storing comprises:

creating a byte cache index segment and a byte cache data segment for each media cache segment; and

storing the byte cache index segments and the byte cache data segments in the cache file.

9. **(Original)** A method as defined in claim 7, wherein the act of storing comprises:

creating a byte cache index segment and a byte cache data segment for each segment; and

serializing the byte cache index segments and the byte cache data segments in the cache file.

10. **(Original)** A method as defined in claim 7, wherein the cache file is stored in a non-volatile manner.

11. **(Proposed Amended)** A system comprising:

a processor;

a data storage module;

a caching module operable to receive and store a plurality of temporally non-contiguous portions of a streaming media file in a cache file in the data storage module, two or more of the plurality of temporally non-contiguous portions being encoded at different bit rates, wherein the caching module comprises processor executable code.

12. **(Original)** A system as defined in claim 11, wherein the data storage module comprises a non-volatile data storage device.

13. **(Proposed Canceled)**

**14. (Proposed Amended)** A system as defined in claim 11, wherein the caching module comprises:

a media cache module operable:

to store each of the plurality of temporally non-contiguous portions as a media cache segment in one of a plurality of media cache streams; and  
parse each media cache segment into a byte cache index segment and a byte cache data segment.

**15. (Original)** A system as defined in claim 11, wherein the caching module comprises:

a media cache module operable to:

store each of the plurality of temporally non-contiguous portions as a media cache segment in one of a plurality of media cache streams, each media cache stream being associated with a different bit rate; and  
parse each media cache segment into a byte cache index segment and a byte cache data segment; and

a byte cache module operable to store the byte cache index segments and the byte cache data segments in the cache file.

**16. (Original)** A system as defined in claim 11, wherein the caching module comprises:

a media cache module operable to:

create a plurality of media cache streams, each media cache stream being associated with a unique bit rate; and

store each temporally non-contiguous portion as a media cache segment in a media cache stream associated with a bite rate at which the temporally non-contiguous portion was encoded; and

parse each media cache segment into a byte cache index segment and a byte cache data segment; and

a byte cache module operable to:

store the byte cache index segments and the byte cache data segments in the cache file.

**17. (Original)** A system as defined in claim 11, wherein the two or more of the plurality of temporally non-contiguous portions include a first video portion encoded at a first bit rate, a second video portion encoded at a second bit rate, and an audio portion, and wherein the first video portion, the second video portion, and the audio portion are stored in different media cache streams.

**18. (Original)** A system as defined in claim 11, wherein:

the streaming media file includes different data types; and

the caching module is operable to:

create a plurality of media cache streams, each media cache stream being associated with a streamed media data type and a streamed media encoded bit rate;

store each temporally non-contiguous portion of received streamed media data in a media cache stream associated with the streamed media data type and a streamed media encoded bit rate of the temporally non-contiguous portion; and

store the media cache streams in the cache file.

- 19. (Original)** A system as defined in claim 11, wherein:
- the streaming media file includes different data types; and
- the caching module is operable to:
- create a plurality of media cache streams, each media cache stream being associated with a streamed media data type and a streamed media encoded bit rate; and
- store each temporally non-contiguous portion of received streamed media data as a media cache segment in a media cache stream associated with the streamed media data type and a streamed media encoded bit rate of the temporally non-contiguous portion;
- parse each media cache segment into a byte cache index segment and a byte cache data segment; and
- store the byte cache index segments and the byte cache data segments in the cache file.

- 20. (Original)** A system as defined in claim 11, wherein the caching module is operable to:

store each of the plurality of temporally non-contiguous portions as a media cache segment in one of a plurality of media cache streams;  
create a segment/stream map specifying the media cache segment and stream in which each temporally non-contiguous portion is stored; and  
parse each media cache segment into a byte cache index segment and a byte cache data segment.

**21. (Proposed Amended)** A computer-readable storage medium having computer-executable instructions for performing acts comprising:

storing at a client a plurality of temporally non-contiguous portions of a streaming media file received from a streaming media source in a cache file, each of the plurality of temporally non-contiguous portions being encoded at a different bit rate, wherein the act of storing comprises:

receiving a first video portion of the streaming media file encoded at a first bit rate;

storing the first video portion in a media cache video stream associated with the first bit rate;

receiving a second video portion of the streaming media file encoded at a second bit rate;

storing the second video portion in a media cache video stream associated with the second bit rate;

receiving a third video portion of the streaming media file encoded at a first bit rate, the a third video portion being temporally non-contiguous from the first video portion;

storing the third video portion in the media cache video stream associated with the first bit rate;

receiving a first audio portion of the streaming media file; and

storing the first audio portion in a media cache audio stream; and

storing the audio and video media cache streams in a cache file.

**22. (Proposed Amended)** A computer-readable storage medium as defined in claim 21, wherein the act of storing comprises:

receiving a first video portion of the streaming media file encoded at a first bit rate;

storing the first video portion in a media cache video stream associated with the first bit rate;

receiving a second video portion of the streaming media file encoded at a second bit rate;

storing the second video portion in a media cache video stream associated with the second bit rate;

receiving a first audio portion of the streaming media file;

storing the first audio portion in a media cache audio stream; and

storing the audio and video media cache streams in a cache file.

23. **(Proposed Canceled)** A computer readable storage medium as defined in claim 21, wherein the act of storing comprises:

receiving a first video portion of the streaming media file encoded at a first bit rate;

storing the first video portion in a media cache video stream associated with the first bit rate;

receiving a second video portion of the streaming media file encoded at a second bit rate;

storing the second video portion in a media cache video stream associated with the second bit rate;

receiving a third video portion of the streaming media file encoded at a first bit rate, the a third video portion being temporally non contiguous from the first video portion;

storing the third video portion in the media cache video stream associated with the first bit rate;

receiving a first audio portion of the streaming media file; and

storing the first audio portion in a media cache audio stream; and

storing the audio and video media cache streams in a cache file.

**24. (Proposed Amended)** A computer-readable storage medium as defined in claim 21, wherein the act of storing comprises:

storing each of the temporally non-contiguous portions in a unique media cache segment;

forming at least two byte cache segments from each media cache segment;

and

storing the byte cache segments in the cache file.

**25. (Proposed Amended)** A computer-readable storage medium as defined in claim 21, wherein the act of storing comprises:

storing each of the temporally non-contiguous portions in at least two byte cache segments; and

storing the byte cache segments in the cache file.

**26-35 (Withdrawn)**

**36. (Original)** A system comprising:

means for receiving a plurality of temporally non-contiguous portions of a streaming media file, at least two of the plurality of temporally non-contiguous portions being encoded at a different bit rate; and

means for associating and storing the plurality of temporally non-contiguous portions in a data structure.